



# Upcoming MediaWiki goodies

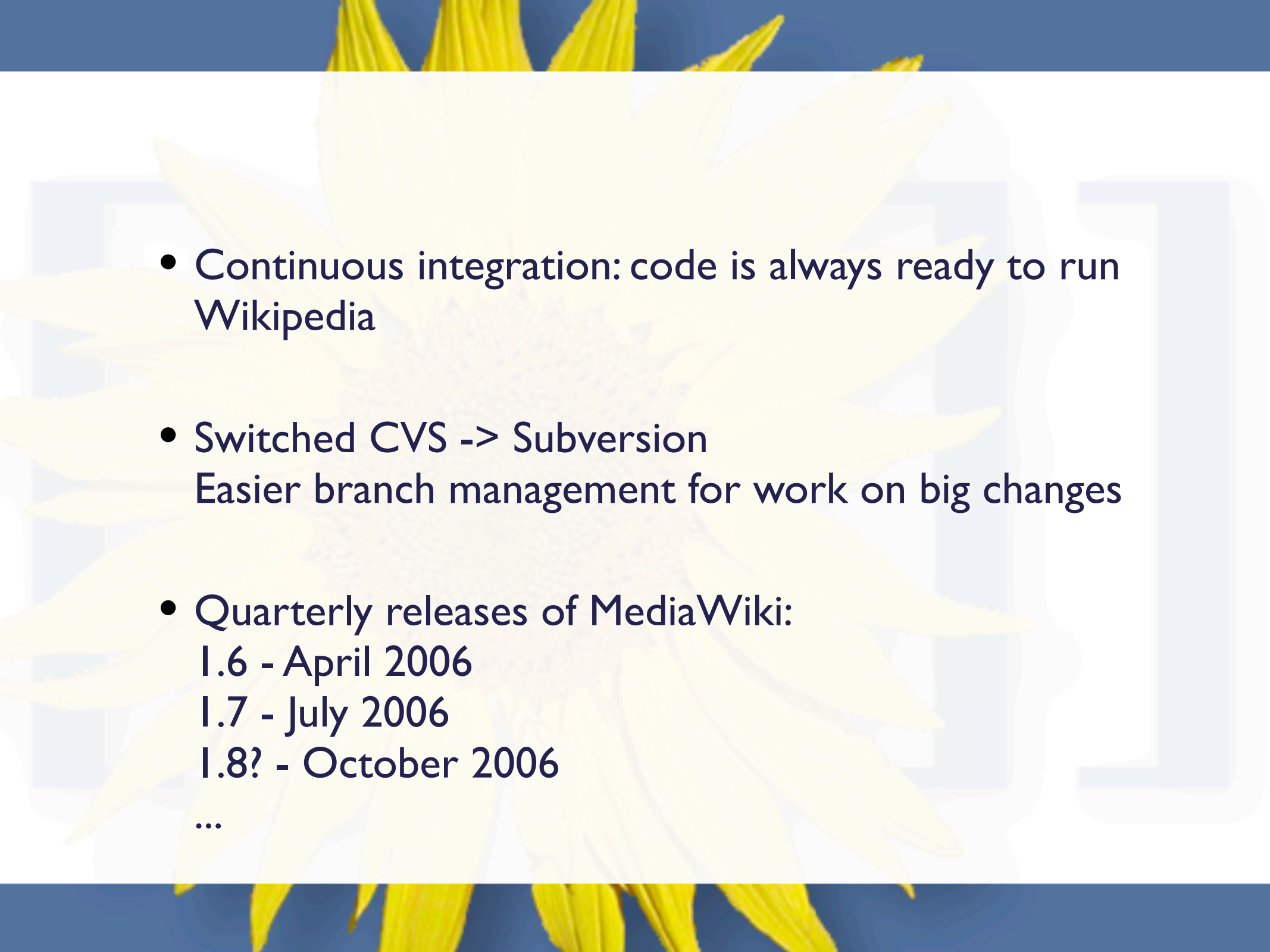
---

(aka, Wikipedia takes over everything)

Brion Vibber  
Wikimedia Foundation, Inc.

April 26, 2006

MySQL Users Conference  
Santa Clara, CA

- 
- Continuous integration: code is always ready to run Wikipedia
  - Switched CVS -> Subversion  
Easier branch management for work on big changes
  - Quarterly releases of MediaWiki:
    - 1.6 - April 2006
    - 1.7 - July 2006
    - 1.8? - October 2006
    - ...

# Where do we go from here?

- **Share the love!**  
You know, all that Web 2.0 crap
- **Make it work!**  
Friendlier failure modes
- **Make it fast!**  
Avoid click-n-wait



# Integration and Outreach

---

Beta bubble buzzword bingo!



# Identity is theft

- Signing up for accounts everywhere sucks!
- Wikipedia and its sister projects to merge user accounts soon
- OpenID, Yadis etc will allow consistent identity on wikis and blogs run by different operators

# Web services API

- Screen-scraping sucks
- Many fun uses for querying data from Wikipedia, Wiktionary, or Wikimedia Commons
- Current work targeting maintenance tools, but content is coming...

# Put the Media in MediaWiki

- Wikimedia Commons stores photos, sound, and video clips for Wikipedia & friends
- GNU FDL, Creative Commons, Public Domain
- Planning a handy service for remote use in off-site wikis and blogs etc
- (Psst... maybe steal flickr's API?)



# Failing gracefully

---

On our budget, two nines sound pretty good



# Some failures are hard

- Meteor strike
- Nuclear war
- Flaky circuit breaker between the UPS and all your machines

# But it can be easy on the eyes

- Our proxy caches in Europe and Asia can serve read-only data to some visitors, some of the time
- Smarter proxy caches should be able to serve read-only data to everyone, transparently, if core servers are unreachable
- With a DNS update on failover we could do this for North America too



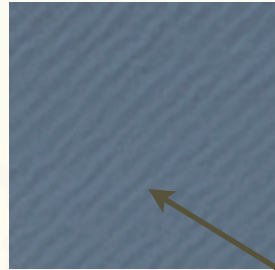
**186,000 miles per second**

---

It's not just a good idea, it's the law!

# War on Latency: Network

Amsterdam



Caching proxies are awesome!

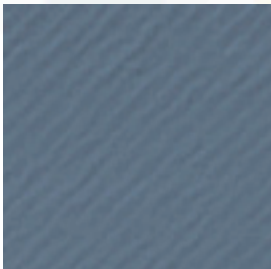
15ms



Frankfurt

- Very fast for static, public page views
- Geographically distributed caches reduce network latency

Tampa

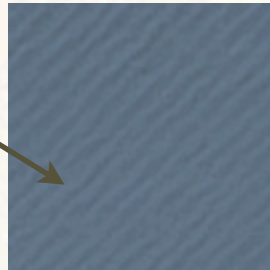
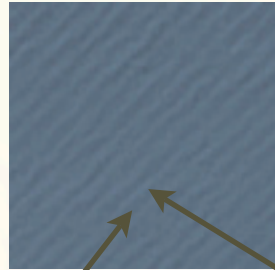




# War on Latency: Network

Amsterdam

BUT proxies are dumb...



15ms

Frankfurt

90ms

Tampa

- Login cookie? Page changed since last view?
- Requests now cross an ocean...
- 100-400ms of dead time, depending on continent. Yuck!

# Vertical partitioning?

- Low-latency access to local read-only database slaves
- Transparently pass write operations up the chain
- Provides read-only view if core servers unreachable

**BUT...**

Freakin' hard to do right.

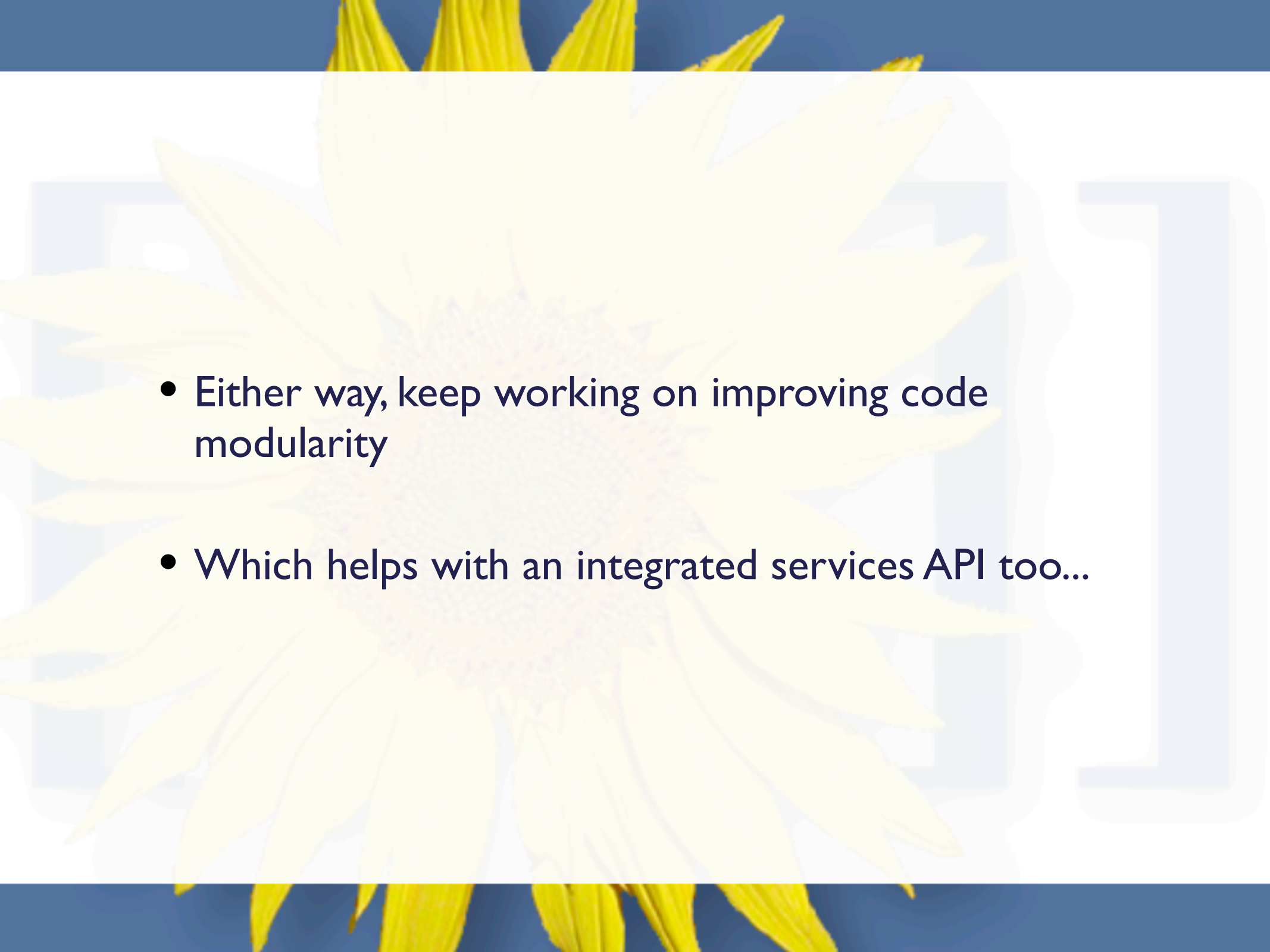
Internal caches need to be kept properly in sync too...

# Aggressive AJAX?

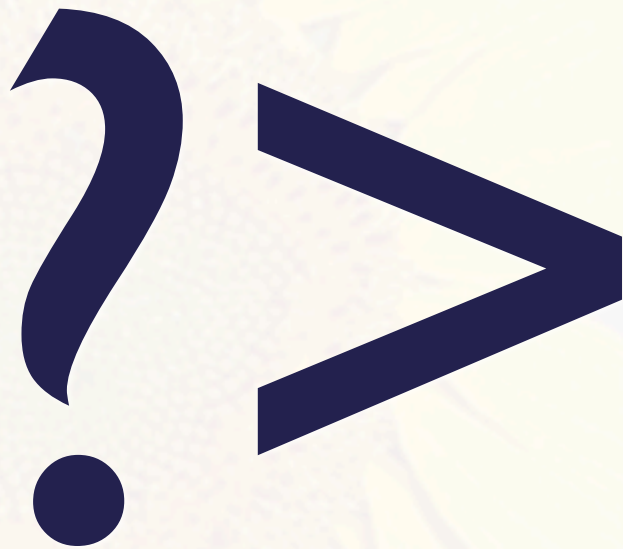
- Run most user login / customization through JavaScript
- Keep most real data in cache-friendly static responses

**BUT...**

Harms accessibility

- 
- Either way, keep working on improving code modularity
  - Which helps with an integrated services API too...





Brion Vibber  
Wikimedia Foundation, Inc.

April 26, 2006

MySQL Users Conference  
Santa Clara, CA